

泰首主控设备对接文档

V2.0.28



深圳泰首智能技术有限公司

版权所有 内部资料 注意保密

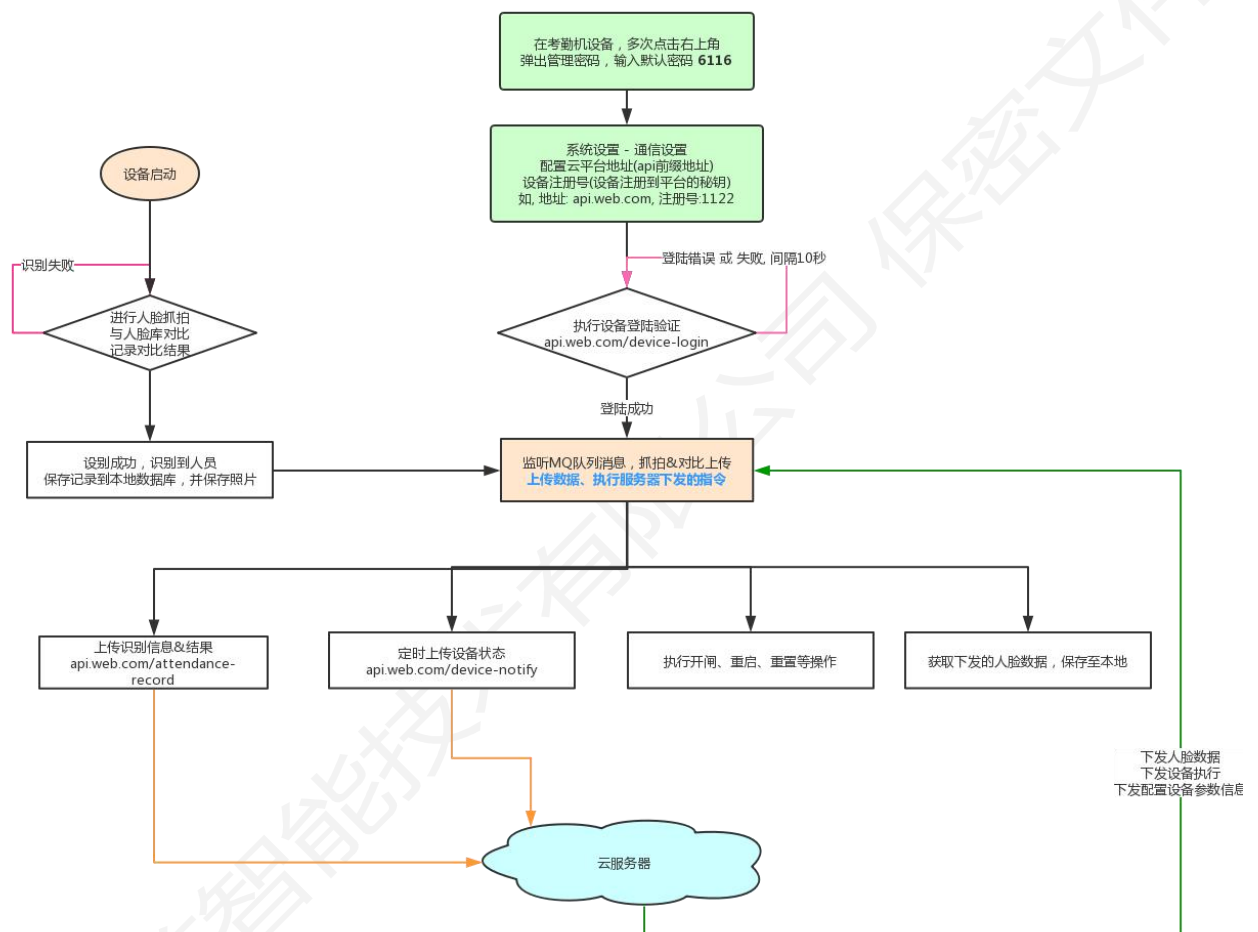
目录

1 接口规则说明	4
1.1 接口调用流程定义图	4
1.2 设备与平台信息通信定义	5
1.3 数据上传、下发流转定义	6
2 设备管理	6
2.1 设备通信配置	6
2.2 设备登陆验证接口	7
2.2.1 设备登录请求	7
2.2.2 设备请求 1:N 比对服务定义	9
2.2.3 设备请求 1:1 比对服务定义	10
2.3 下发设备操作至 MQ (RABBITMQ)	11
2.3.1 MQ 服务安装	11
2.3.2 MQ 通信示例、使用文档	11
2.3.3 下发设备开门命令	11
2.3.4 下发重置人员与历史识别记录命令	12
2.3.5 下发重置人员数据命令	13
2.3.6 下发重置历史识别记录命令	14
2.3.7 下发重启设备命令	14
3 人员管理	15
3.1 下发人员信息至 MQ (RABBITMQ)	15
3.1.1 平台下发新增、修改、删除信息至 MQ	15

3.1.2 设备获取新增、修改人员信息.....	16
3.1.3 名单注册模式，信息上传注册.....	18
4 上传信息.....	20
4.1 上传人脸比对/识别信息.....	20
4.2 上传设备状态信息.....	22
5 广告管理.....	23
5.1 设备广告初始化.....	23
5.2 下发、调整广告信息至 MQ 设备（RABBITMQ）	24
5.1.1 下发广告更新信息至 MQ.....	24
5.1.2 获取广告信息、更新广告.....	25

1 接口规则说明

1.1 接口调用流程定义图



~~备注:

设备通过设备中的配置的: 通信地址+配置的注册号+设备序列号, 执行登陆请求

通信地址: 执行登陆、人员信息获取、识别上传等信息通信的服务器接口地址

机构码: 可按需配置, 用于多公司平台, 区分设备所在公司、区域等信息

设备序列号: 不可修改, 设备生成唯一序列号, 用于判断设备唯一性, MQ 队列通过该值定义队列名进行通信

如设备配置的接口地址为: api.web.com

注册号为: 9527

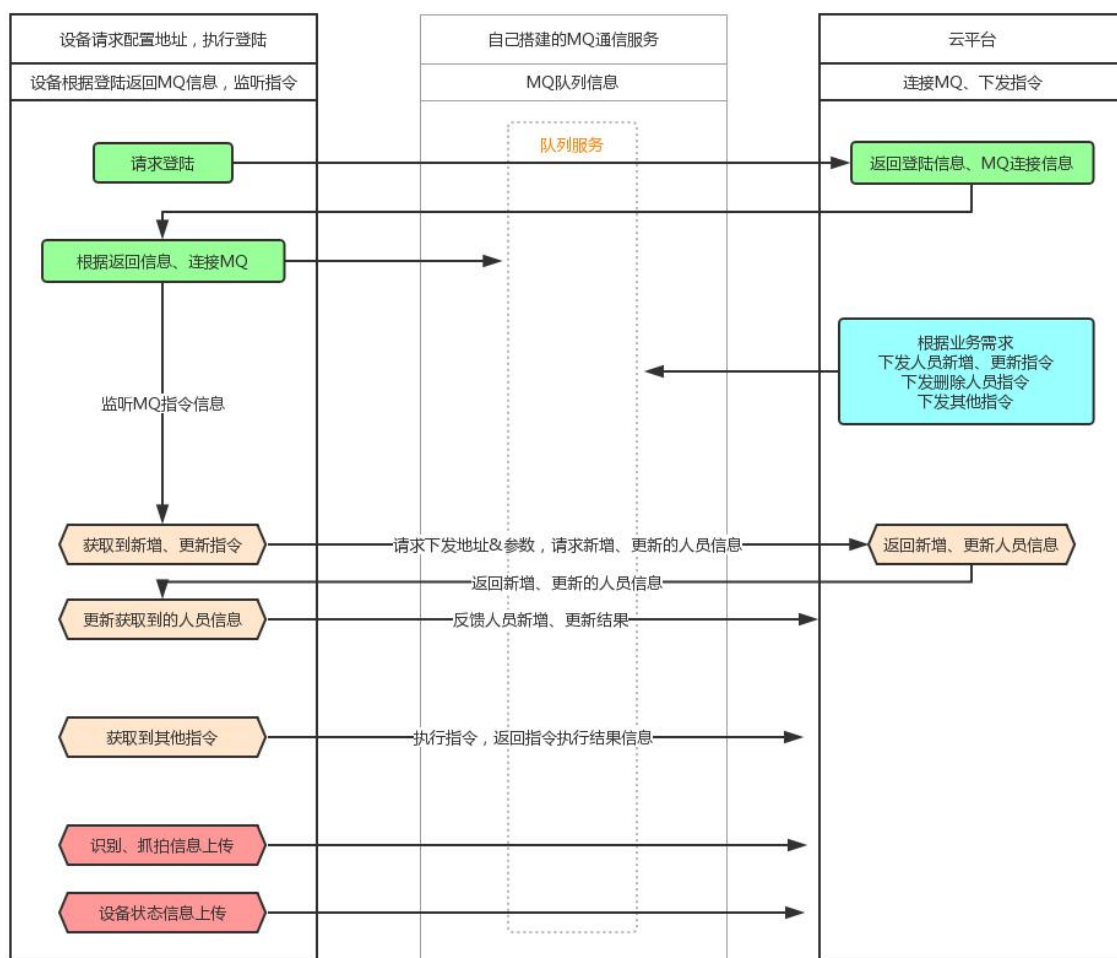
设备序列号为: DGJKLOAUEK129301NDD

则登陆请求地址为:

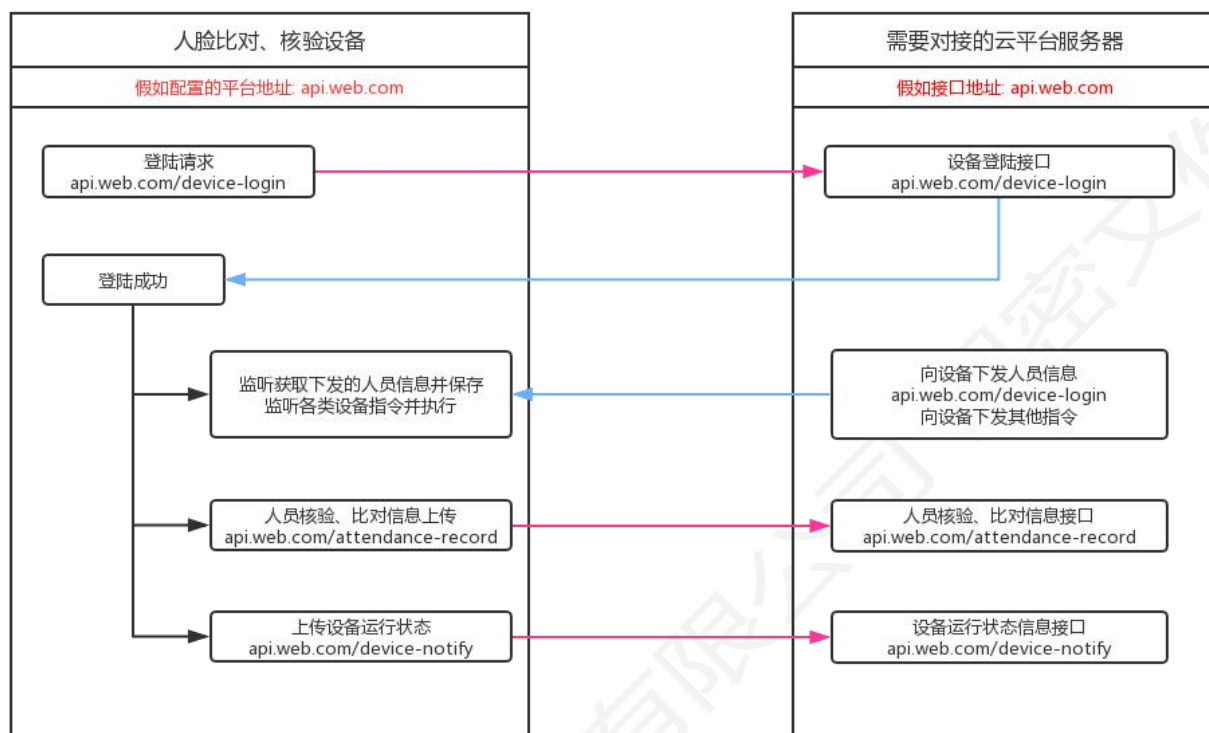
http://api.web.com/device-login?sign_type=register&sign_code=9527&device_sn=DGJKLOAUEK129301NDD

1.2 设备与平台信息通信定义

MQ存在原因: 当设备部署在局域网内, 外网云平台无法直接向设备进行通信, 因此使用MQ队列消息, 平台向MQ推送指令, 设备连接MQ获取到指令, 执行对应操作



1.3 数据上传、下发流转定义



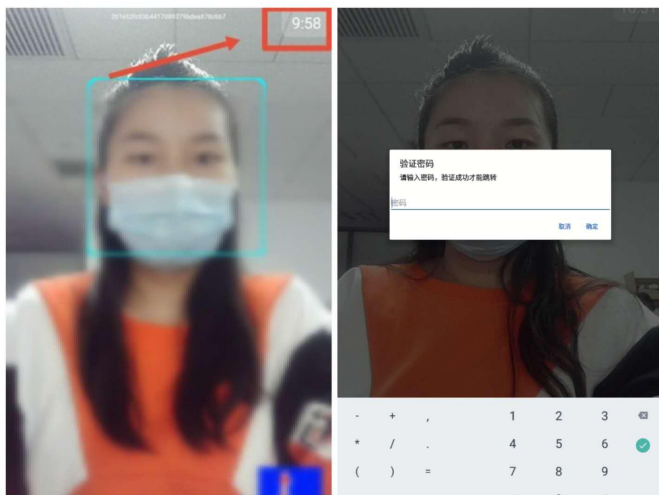
2 设备管理

设备类型包括安卓身证通、访客机、注册机、迎宾机

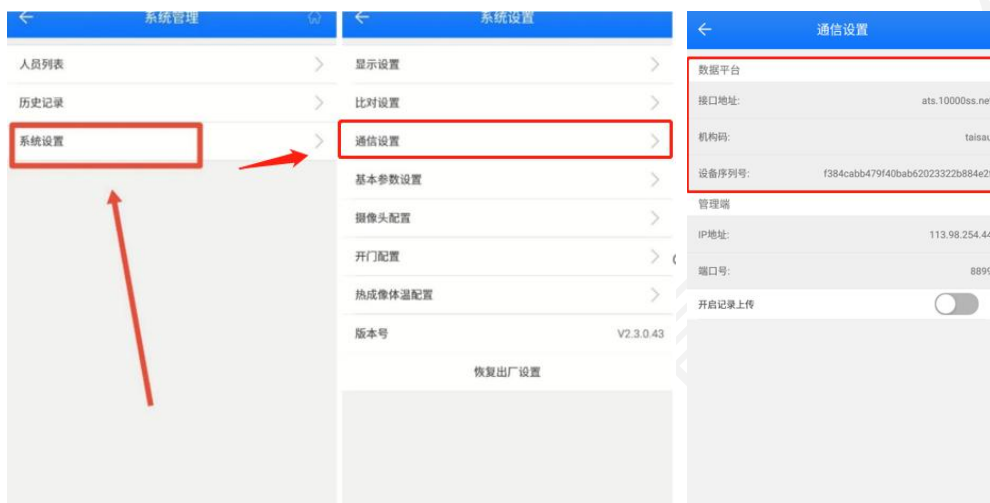
2.1 设备通信配置

设备在使用前需要配置对接平台地址，具体步骤如下：

- ① 开机后，主界面显示如下，连续点击 5 次右上角的时间，弹出密码输入窗口，在密码输入框中输入**默认密码: 6116**；



② 在系统管理界面，点击 系统设置-通信设置，进入对接平台设置界面，如图所示：



- 对接仅需要配置数据平台地址
- 接口地址：平台接口地址，可以是内网 ip 地址。如 192.168.1.11/device
- 设备登录地址默认为：接口地址/device-login。如 192.168.1.11/device/device-login
- 机构码：设备登录会将此参数传递至登录接口，可以用于区分设备所在区域，非必须。
- 设备序列号：设备唯一序列号，不可修改，可根据此参数绑定设备信息，后续 MQ 信息下发以此命名队列名称。

2.2 设备登陆验证接口

2.2.1 设备登录请求

- 设备启动后首先向对接平台执行设备登陆请求，请求成功后才会执行后续操作
- 接口地址在设备配置中进行配置，调试时可设定为本地服务 IP
- 设备只能在管理平台由管理员进行注册。注册并授予权限的设备才可以登陆服务器。
- 设备登陆平台返回 token，device_sn，后续所有接口请求都会返回该信息用作校验字段，同时 token 会作为 Authorization 在 header 加入请求信息
- token 对于正在通信的设备不需要更新，如果设备未通信超过 30 分钟则更新 token，当设备带旧的 token 到平台，可

返回 40004, 提示设备重新登陆

接口用途	设备登陆				
接口地址	http://api.example.com/device-login				
请求方式	POST				
请求格式	application/json				
头部	参数	必填	类型	说明	值
	X-Requested-With	Y	string	请求类型	XMLHttpRequest
	X-Domain-With	Y	string	请求域	Api
请求	参数	必填	类型	说明	示例值
	sign_type	Y	string	签名类型	register
	sign_code	Y	string	机构码	9527
	device_sn	Y	string	设备序列号/唯一标识	A48072C64BC043BBAC7EC95F22400075
	device_type	Y	string	设备类型	taisau_window、taisau_android、taisau_cloud
	device_software		string	软件名称	facepass
	device_version		string	软件版本	V1.2.3
	login_mode		string	登录模式,	init(安装后首次登陆) / login(后续普通登录)
返回	参数	必填	类型	说明	示例值
	success	Y	boolean	状态	true
	code	Y	int	状态码	0
	message	Y	string	返回信息	登陆成功
	data	Y	object	设备信息	{}
	device_sn	Y	string	设备序列号	A48072C64BC043BBAC7EC95F22400075
	token	Y	string	登陆 token	\$2y\$10\$4lkYfHXv03GzRNfW/poWq...
	score		float	识别推送阈值	82
	alive		boolean	活体检测	false
	attendanceScore		float	考勤、开门分数	根据设备而异, 默认值为 82
	recordAddr		string	记录上传地址	默认为: 配置地址+/attendance-record
	recordUploadScore		float	记录上传最小分数	根据设备而异, 默认值为 0
	recordUploadPeriod		int	记录上传间隔/秒	根据设备而异, 默认实时上传
	recordUploadMode		int	记录上传模式	0:单记录上传, 1-10:多记录上传,最多上传数
	recordUploadReg		int	是否上传模版照	0 否(默认), 1 是(上传)
	deviceMonitorAddr		string	设备状态上报地址	默认为: 配置地址+/device-notify
	deviceMonitorPeriod		int	设备状态上报间隔	默认为 120 秒
	advertiseAddr		string	广告展示请求地址	默认为: 配置地址+/attendance-advertise
	advertisePeriod		int	广告更新间隔	根据设备而异, 默认只在设备启动请求
	rabbitmq	Y	object	Rabbitmq 连接信息	{}
	>> exchange	Y	object	Rabbitmq 交换机信息	{}
	qrcode		object	二维码信息	{}, 设备 V2.3.0.28 必须
	>> visitor		string	访客二维码	
	compareServer		object	比对服务器	若返回此参数,设备将在未识别后调用比对服务
	>> compare		string	1:N 请求地址	http://api.example.com/compare
	>> compareOne		string	1:1 比对地址	http://api.example.com/compare-one
	>> params		object	请求比对服务参数	{ token: '112233' }

返回示例					
	<pre>{ "success": true, "code": 0, "message": null, "data": { "name": "考勤云台", "address": "泰首公司内部" }, "device_sn": "A48072C64BC043BBAC7EC95F22400075", "token": "\$2y\$10\$4lkYfHXv03GzRNfW/poWqeQApfrkaEfizQEFa3aX.LQqagc9.l48q", "score": 82, "alive": false, "rabbitmq": { "host": "mq.chinaanno.com", "port": "5672", "vhost": "/", "login": "taisau", "password": "taisau", "queue": "A48072C64BC043BBAC7EC95F22400075", "exchange": { "name": "taisau-mq-exchange-facenew", "type": "topic", "passive": false, "durable": true, "auto_delete": false, "arguments": null }, "qrcode": { "public": "", "applet": "", "visitor": "" } } }</pre>				

2.2.2 设备请求 1:N 比对服务定义

- 仅在设备版本 2.3.42 以后版本支持
- 根据登录返回而定，如登录返回 `compareServer` 参数，当设备识别不成功时，会请求该服务进行二次比对
- 请求参数会与 { capture: [base64] } 合并后请求

接口用途	设备比对服务，接口地址根据登录返回信息而定
接口地址	http://api.example.com/compare?device_sn={device_sn}&token={token}
请求方式	POST

请求格式	application/json				
头部	参数	必填	类型	说明	值
	X-Requested-With	Y	string	请求类型	XMLHttpRequest
	X-Domain-With	Y	string	请求域	Api
	Authorization	Y	string	验证信息	Bearer + [登陆返回的 token]
请求	参数	必填	类型	说明	示例值
	capture	Y	base64	现场抓拍照片	
	...params			比对附加参数	与 { capture: [base64] } 合并
返回	参数	必填	类型	说明	示例值
	success	Y	boolean	状态	true
	code	Y	int	状态码	0
	message	Y	string	返回信息	识别成功
	data	Y	array	比对成功列表	按分值排序, 0-100
返回示例					
	<pre>{ "success": true, "code": 0, "message": null, "data": [{ personId: "身份 ID" , score: 98.8 }, { personId: "身份 ID" , score: 68.8 }, ] }</pre>				

2.2.3 设备请求 1:1 比对服务定义

- 仅在设备版本 2.3.42 以后版本支持
- 根据登录返回而定，如登录返回 `compareServer` 参数，当设备识别不成功时，会请求该服务进行二次比对
- 请求参数会与 { capture: [base64] } 合并后请求

接口用途	设备比对服务, 接口地址根据登录返回信息而定				
接口地址	http://api.example.com/compare-one?device_sn={device_sn}&token={token}				
请求方式	POST				
请求格式	application/json				
头部	参数	必填	类型	说明	值
	X-Requested-With	Y	string	请求类型	XMLHttpRequest

	X-Domain-With	Y	string	请求域	Api
	Authorization	Y	string	验证信息	Bearer + [登陆返回的 token]
请求	参数	必填	类型	说明	示例值
	capture	Y	base64	抓拍照片	
	other	Y	base64	比对照片	
	...params			比对附加参数	与{ capture: [base64] } 合并
返回	参数	必填	类型	说明	示例值
	success	Y	boolean	状态	true
	code	Y	int	状态码	0
	message	Y	string	返回信息	识别成功
	data	Y	array	比对成功列表	按分值排序, 0-100
返回示例	<pre>{ "success": true, "code": 0, "message": null, "data": { score: 98.8 } }</pre>				

2.3 下发设备操作至 MQ (RabbitMQ)

2.3.1 MQ 服务安装

<https://www.rabbitmq.com/download.html>

安装好后需在 MQ 管理后台添加 自动定义名称为: rabbit-mq-exchange 的交换机

2.3.2 MQ 通信示例、使用文档

- PHP: <https://www.rabbitmq.com/tutorials/tutorial-one-php.html>
- JAVA: <https://www.rabbitmq.com/tutorials/tutorial-one-java.html>
- Python: <https://www.rabbitmq.com/tutorials/tutorial-one-python.html>
- C#: <https://www.rabbitmq.com/tutorials/tutorial-one-dotnet.html>
- Javascript: <https://www.rabbitmq.com/tutorials/tutorial-one-javascript.html>

2.3.3 下发设备开门命令

- 平台通过 MQ 下发信息到设备, 下发时以 device_sn 命名队列名称, 设备通过该队列名称获取下发消息。

- 设备根据 method 进行命令识别

信息说明	下发设备开门 MQ 信息				
信息用途	对设备执行开门操作				
下发数据	参数	必填	类型	说明	示例值
	method	Y	string	下发方式	open-door
	data	Y	object	下发数据	{}
	notify	Y	string	执行结果通知地址	http://api.example.com/device-notify
	params	Y	object	请求 post 数据	{}
下发示例	<pre>{ method: "open-door" , data: {}, notify : "" , params: {} }</pre>				
反馈头部	参数	必填	类型	说明	值
	X-Requested-With	Y	string	请求类型	XMLHttpRequest
	X-Domain-With	Y	string	请求域	Api
	Authorization	Y	string	验证信息	Bearer + [登陆返回的 token]
下发反馈	notify 字段如果有值，则反馈下发状态给 notify 地址				
反馈地址	http://api.example.com/device-notify?device_sn={device_sn}&token={token}				
	<pre>{ method: "open -door" , params: {}, success: false, message: "权限不足!" }</pre>				

2.3.4 下发重置人员与历史识别记录命令

- 平台通过 MQ 下发信息到设备，下发时以 device_sn 命名队列名称，设备通过该队列名称获取下发消息。
- 设备根据 method 进行命令识别

信息说明	下发重置指令到 MQ				
信息用途	对设备人员、历史进行重置操作				
下发数据	参数	必填	类型	说明	示例值
	method	Y	string	下发方式	device-reset
	data	Y	object	下发数据	{}
	notify	Y	string	执行结果通知地址	http://api.example.com/device-notify
	params	Y	object	请求 post 数据	{}
下发示例	<pre>{ method: "device-reset" , }</pre>				

	<pre>data: {}, notify: "", params: {} }</pre>				
反馈头部	参数	必填	类型	说明	值
	X-Requested-With	Y	string	请求类型	XMLHttpRequest
	X-Domain-With	Y	string	请求域	Api
	Authorization	Y	string	验证信息	Bearer + [登陆返回的 token]
下发反馈	notify 字段如果有值, 则反馈下发状态给 notify 地址				
反馈地址	http://api.example.com/device-notify?device_sn={device_sn}&token={token}				
	<pre>{ method: "device -reset" , params: {}, success: false, message: "不支持此命令!" }</pre>				

2.3.5 下发重置人员数据命令

- 平台通过 MQ 下发信息到设备, 下发时以 device_sn 命名队列名称, 设备通过该队列名称获取下发消息。
- 设备根据 method 进行命令识别

信息说明	下发重置指令到 MQ				
信息用途	对设备人员进行重置操作				
下发数据	参数	必填	类型	说明	示例值
	method	Y	string	下发方式	reset-person
	data	Y	object	下发数据	{}
	notify	Y	string	执行结果通知地址	http://api.example.com/device-notify
	params	Y	object	请求 post 数据	{}
下发示例	<pre>{ method: "reset-person" , data: {}, notify: "", params: {} }</pre>				
反馈头部	参数	必填	类型	说明	值
	X-Requested-With	Y	string	请求类型	XMLHttpRequest
	X-Domain-With	Y	string	请求域	Api
	Authorization	Y	string	验证信息	Bearer + [登陆返回的 token]
下发反馈	notify 字段如果有值, 则反馈下发状态给 notify 地址				
反馈地址	http://api.example.com/device-notify?device_sn={device_sn}&token={token}				
	<pre>{ method: "reset-person" ,</pre>				

```

params: {},
success: false,
message: "重置完成!"
}

```

2.3.6 下发重置历史识别记录命令

- 平台通过 MQ 下发信息到设备，下发时以 device_sn 命名队列名称，设备通过该队列名称获取下发消息。
- 设备根据 method 进行命令识别

信息说明	下发重置指令到 MQ				
信息用途	对设备历史进行重置操作				
下发数据	参数	必填	类型	说明	示例值
	method	Y	string	下发方式	reset-history
	data	Y	object	下发数据	{}
	notify	Y	string	执行结果通知地址	http://api.example.com/device-notify
	params	Y	object	请求 post 数据	{}
下发示例	<pre> { method: "reset-history ", data: {}, notify: "", params: {} } </pre>				
反馈头部	参数	必填	类型	说明	值
	X-Requested-With	Y	string	请求类型	XMLHttpRequest
	X-Domain-With	Y	string	请求域	Api
	Authorization	Y	string	验证信息	Bearer +[登陆返回的 token]
下发反馈	notify 字段如果有值，则反馈下发状态给 notify 地址				
反馈地址	http://api.example.com/device-notify?device_sn={device_sn}&token={token}				
	<pre> { method: "reset-history" , params: {}, success: false, message: "重置完成!" } </pre>				

2.3.7 下发重启设备命令

- 平台通过 MQ 下发信息到设备，下发时以 device_sn 命名队列名称，设备通过该队列名称获取下发消息。
- 设备根据 method 进行命令识别

信息说明	下发重启指令到 MQ
------	------------

信息用途	对设备执行重启操作				
下发数据	参数	必填	类型	说明	示例值
	method	Y	string	下发方式	device-restart
	data	Y	object	下发数据	{}
	notify	Y	string	执行结果通知地址	http://api.example.com/device-notify
	params	Y	object	请求 post 数据	{}
下发示例	<pre>{ method: "reset-history ", data: {}, notify: "", params: {} }</pre>				
反馈头部	参数	必填	类型	说明	值
	X-Requested-With	Y	string	请求类型	XMLHttpRequest
	X-Domain-With	Y	string	请求域	Api
	Authorization	Y	string	验证信息	Bearer + [登陆返回的 token]
下发反馈	notify 字段如果有值，则反馈下发状态给 notify 地址				
反馈地址	http://api.example.com/device-notify?device_sn={device_sn}&token={token}				
	<pre>{ method: "reset-history" , params: {}, success: false, message: "重启执行完成!" }</pre>				

3 人员管理

3.1 下发人员信息至 MQ (RabbitMQ)

- 平台通过 MQ 下发人员信息到设备
- 终端获取消息后将发送新增、修改、删除人员的相关请求。参数是以 fromdata 形式上传
- 照片必须小于 500KB
- 删除人员会直接进行人员删除，不再请求接口
- 人员请求地址通过 MQ 下发的 path 地址决定，params 会原样 post 至该地址

3.1.1 平台下发新增、修改、删除信息至 MQ

- 平台通过 MQ 下发信息到设备，下发时以 device_sn 命名队列名称，设备通过该队列名称获取下发消息。
- 设备根据 method 进行命令识别

信息说明	下发人员信息，设备接收到下发的信息请求平台更新人员信息
------	-----------------------------

信息用途	新增、修改、删除 人员信息				
下发数据	参数	必填	类型	说明	示例值
	method	Y	string	下发方式	sync-person 新增、修改 / delete-person 删除
	person_list	Y	array	下发人员 ID 列表	[12306,12307,12308...]
	path	Y	string	请求人员信息地址	http://api.example.com/person-list
	notify	Y	string	执行结果通知地址	http://api.example.com/person-notify
	params	Y	object	请求 post 数据	{ limit: 100, size: 100 }
下发示例	<pre>{ method: "sync -person" , person_list: [12306,12307,12308], path : "http://api.example.com/push-identity" , notify: "http://api.example.com/push-identity-notify" , params: { limit: 100, size: 100 } }</pre>				
反馈头部	参数	必填	类型	说明	值
	X-Requested-With	Y	string	请求类型	XMLHttpRequest
	X-Domain-With	Y	string	请求域	Api
	Authorization	Y	string	验证信息	Bearer +[登陆返回的 token]
下发反馈	notify 字段如果有值，则反馈下发状态给 notify 地址				
反馈地址	http://api.example.com/person-notify?device_sn={device_sn}&token={token}				
	<pre>{ method: "sync -person" , params: {}, success: true, message: "人员下发成功!" successful: [12306, 12307], failed:[{ person_id: 12308, reason: "图片人脸尺寸太小" }] }</pre>				

3.1.2 设备获取新增、修改人员信息

- 请求地址通过 MQ 下发的 path 地址决定，params 会原样 post 至该地址
- 每次下发人员信息建议不超过 50 人，照片必须小于 500KB

接口用途	返回人员信息				
接口地址	MQ 下发的 path 地址，例如: http://api.example.com/push-identity?device_sn={device_sn}&token={token}				
请求方式	POST				
请求格式	application/json				
请求	参数	必填	类型	说明	示例值
	person_list	Y	Array	MQ 下发的参数	[12306,12307,12308]
	limit		string	MQ 下发的参数	100
	size		string	MQ 下发的参数	100

请求头部	参数	必填	类型	说明	值
	X-Requested-With	Y	string	请求类型	XMLHttpRequest
	X-Domain-With	Y	string	请求域	Api
	Authorization	Y	string	验证信息	Bearer + [登陆返回的 token]
返回	参数	必填	类型	说明	示例值
	success	Y	boolean	状态	true
	code	Y	int	状态码	0
	message	Y	string	返回信息	获取成功
	person_list	Y	array	下发的人员信息列表	{}
	➢ person_id	Y	int	人员 ID	12306
	➢ person_name	Y	string	人员信息	张三
	➢ person_type	Y	string	人员类型	1 白名单, 2 黑名单, 3 访客, 4 VIP, 5 陌生人
	➢ face_list	Y	array	人脸信息 1-3 张	[{img_url: "", img_base64: "", face_id: "" }, ...] img_url 和 img_base64 二选一, 优先 img_base64
	➢ id_card		string	身份证号	
	➢ ic_card		string	IC 卡号	
	➢ sex		string	性别	男, 女
	➢ birthday		string	出生日期	
	➢ valid_time		object	通行时段限制	{ start_time: '2019-01-01 01:11:11', end_time: "", count:10 } 次数: 0 无限次 或 指定次数
	➢ group_id		string	分组 ID	
	➢ group_name		string	分组名	
	➢ group_type		string	分组类型	
返回示例	<pre>{ "success": true, "code": 0, "message": null, "person_list": [{ "birthday": "2019-03-28", "id_card": "", "sex": 1, "person_id": 12880, "person_name": "梁日泉", "group_type": 4, "group_id": 457, "group_name": "体验组", "face_list": [{ "img_url": null, "img_base64": null, "face_id": 123459 }, { "img_url": null, "img_base64": null, "face_id": 123460 }] }</pre>				

```

    }}
  }, {
    "birthday": "2019-04-01",
    "id_card": "",
    "sex": 2,
    "person_id": 12882,
    "person_name": "小邵",
    "group_type": 4,
    "group_id": 457,
    "group_name": "体验组",
    "face_list": [{
      "img_url": null,
      "img_base64": null,
      "face_id": 123460
    }, {
      "img_url": null,
      "img_base64": null,
      "face_id": 123461
    }]
  }]
}

```

3.1.3 名单注册模式，信息上传注册

- 在设备配置 系统设置-比对设置-通行模式中，可将设备调整为注册模式。
- 注册模式下仅能通讯刷身份证进行人脸比对，人证比对一致后，将传递信息至记录上传接口，并附加 register 参数
- 可在信息接口通信 register 参数进行信息判断，进行人脸注册

接口用途	注册模式，上传身份证注册信息				
接口地址	http://api.example.com/attendance-record?register=true&device_sn={device_sn}&token={token}&device_type=taisau_android				
请求方式	POST				
请求格式	application/json				
反馈头部	参数	必填	类型	说明	值
	X-Requested-With	Y	string	请求类型	XMLHttpRequest
	X-Domain-With	Y	string	请求域	Api
	Authorization	Y	string	验证信息	Bearer + [登陆返回的 token]
请求	参数	必填	类型	说明	示例值
	register	Y	boolean	注册模式	true
	person_id	Y	int	人员 ID	
	person_name	Y	string	人员姓名	
	person_type	Y	int	人员类型	
	direction	Y	int	设备进出方向	0 进，1 出，2 进&出
	equipmentName			设备名称	

	id_card		string	身份证号码	
	ic_card		string	IC 卡号	
	card_type		int	证件类型	0: 人脸识别, 1: 身份证, 2: IC 卡 3: 港澳台居住证, 4: 中国护照 5: 国外护照, 6: 永久居住证, 100: 其他
	card_data		object	证件数据	读取的证件数据, 身份证信息、IC 卡信息等
	➤ number		string	证件号码	
	➤ name		string	证件姓名	
	➤ picture		string	证件照片	
	➤ sex		int	性别	0 未知 1 男性 2 女性
	➤ nation		string	民族	
	➤ birthday		string	生日	
	➤ address		string	地址	
	➤ vaild_start		string	有效期-开始	2019-01-01
	➤ vaild_end		string	有效期-结束	2030-01-01
	➤ department		string	发卡机构	
	face_img		base64	人脸照片	
	capture_img	Y	base64	抓拍照片	
	capture_time	Y	string	抓拍时间	
	capture_score	Y	string	抓拍分数	
	capture_status	Y	int	抓拍对比状态	0 比对通过, 1 比对失败, 2 无模版, 3 无效卡 4 无现场照, 5 非活体, 6 证件过期 7 超过时间限制, 8 超过此数限制 100 未知错误
	temperature		string	检测体温值	36.8
	failed_reason		string	失败原因	比对失败 (人脸比对失败) 无模版 (刷脸无模版) 无效卡 (刷卡无证件信息) 无现场照 (刷卡未抓到照片) 非活体 (人脸抓拍非活体)
返回	参数	必填	类型	说明	示例值
	success	Y	boolean	状态	true
	code	Y	int	状态码	0
	message	Y	string	返回信息	上传成功
	data	Y	array	成功数据	{}
请求示例	{ "capture_img ": "/9j/4AAQSkZJRgABb.....", "capture_score": "85.64", "capture_status": "0" "capture_time": "2018-02-01 13:52:16", "card_type": "1 ", "direction": "2", "equipmentName": "身证通门禁访客系统", "face_img": "", "failed_reason": "比对通过",				

	<pre> "id_card": "420983198902013571", "person_name": "张三", "person_type": "5", card_data: { "address": "湖北省广水市骆店镇石何村", "birthday": "19890201", "department": "广水市公安局", "name": "刘芬", "nation": "汉", "number": "420983198902013571", "picture": "/9j/4AAQSkZJRgABb.....", "sex": "2", "valid_end": "2037.02.04", "valid_start": "2017.02.04" } </pre>
返回示例	<pre> { "code": "0", "data": "", "message": "Register Success!", "success": True, } </pre>

4 上传信息

4.1 上传人脸比对/识别信息

- 接口地址在设备配置中进行配置
- 抓拍分数为与之最相似的人的相似度，后台需要判断分数大于特定值为识别成功（建议 80 以上）
- 低于特定值判断为未识别，陌生人信息

接口用途	上传人员识别信息				
接口地址	http://api.example.com/attendance-record?device_sn={device_sn}&token={token}				
请求方式	POST				
请求格式	application/json				
反馈头部	参数	必填	类型	说明	值
	X-Requested-With	Y	string	请求类型	XMLHttpRequest
	X-Domain-With	Y	string	请求域	Api
	Authorization	Y	string	验证信息	Bearer + [登陆返回的 token]
请求	参数	必填	类型	说明	示例值
上传模式	登陆返回记录上传模式(recordUploadMode)				

	如果默认 recordUploadMode=0, 以记录下放数据的格式进行上传如果 recordUploadMode 为 1-10, 已 capture_list 命名的数据数组进行上传 (下方数据为每个数组的内容)				
	person_id	Y	int	人员 ID	
	person_name	Y	string	人员姓名	
	person_type	Y	int	人员类型	
	direction	Y	int	设备进出方向	0 进, 1 出, 2 进&出
	id_card		string	身份证号码	
	ic_card		string	IC 卡号	
	card_type		int	证件类型	0: 人脸识别, 1: 身份证, 2: IC 卡 3: 港澳台居住证, 4: 中国护照 5: 国外护照, 6: 永久居住证, 100: 其他
	card_data		object	证件数据	读取的证件数据, 身份证信息、IC 卡信息等
	➤ number		string	证件号码	
	➤ name		string	证件姓名	
	➤ picture		string	证件照片	
	➤ sex		int	性别	0 未知 1 男性 2 女性
	➤ nation		string	民族	
	➤ birthday		string	生日	
	➤ address		string	地址	
	➤ valid_start		string	有效期-开始	2019-01-01
	➤ valid_end		string	有效期-结束	2030-01-01
	➤ department		string	发卡机构	
	face_img	Y	base64	人脸照片	
	capture_img	Y	base64	抓拍照片	
	capture_time	Y	string	抓拍时间	
	capture_score	Y	string	抓拍分数	
	capture_status	Y	int	抓拍对比状态	0 比对通过, 1 比对失败, 2 无模版, 3 无效卡 4 无现场照, 5 非活体, 6 证件过期 7 超过时间限制, 8 超过此数限制 100 未知错误
	temperature		string	检测体温值	36.8
	failed_reason		string	失败原因	比对失败 (人脸比对失败) 无模版 (刷脸无模版) 无效卡 (刷卡无证件信息) 无现场照 (刷卡未抓拍到照片) 非活体 (人脸抓拍非活体)
返回	参数	必填	类型	说明	示例值
	success	Y	boolean	状态	true
	code	Y	int	状态码	0
	message	Y	string	返回信息	上传成功
	data	Y	array	成功数据	{}
请求示例					
	{ "person_id": "12306", "person_name": "张三", "person_type": "1",				

	<pre> "id_card": "37062019631013301X", "ic_card": "AA2D1FD1", "card_type": "1 ", "face_img": "13_52_16_628_646821.jpg", "capture_img": "646821_0.jpg", "capture_score": "85.64", "capture_time": "2018-02-01 13:52:16", "capture_status": "1" } </pre>

4.2 上传设备状态信息

- 接口地址在设备配置中进行配置

接口用途	上传实时设备信息				
接口地址	http://api.example.com/device-notify?device_sn={device_sn}&token={token}				
请求方式	POST				
请求格式	application/json				
头部	参数	必填	类型	说明	值
	X-Requested-With	Y	string	请求类型	XMLHttpRequest
	X-Domain-With	Y	string	请求域	Api
	Authorization	Y	string	验证信息	Bearer +[登陆返回的 token]
请求	参数	必填	类型	说明	示例值
	device_ip	Y	string	设备 IP 地址	
	device_name	Y	string	设备名称	
	device_status	Y	string	设备状态	1 正常, 2 停止/待机
	device_software		string	软件名称	facepass
	device_version		string	软件版本	V1.2.3
	device_address		string	设备地址	大门口
	gps_location		string	GPS 经纬度	部分设备支持才有此参数 longitude,latitude
	gps_address		string	GPS 解析地址	深圳市南山区
	params		object	设备参数	{ devicename: "taisau_x5", ... }
	direction	Y	int	设备进出方向	0 进, 1 出, 2 进&出
	sdk_version	Y	string	算法版本	
	cpu_usage	Y	string	CPU 使用率	
	memory_size	Y	string	内存大小	
	memory_usage	Y	string	内存使用率	
	disk_size	Y	string	磁盘大小	
	disk_usage	Y	string	磁盘使用率	
	disk_free_size	Y	string	磁盘可用空间	
	camera_list		array	连接相机信息	
	➤ camera_ip		string	相机 IP 地址	

	➤ <i>camera_name</i>		<i>string</i>	相机名称	
	➤ <i>camera_status</i>		<i>string</i>	相机状态	1 正常, 2 停止/待机
返回	参数	必填	类型	说明	示例值
	success	Y	boolean	状态	true
	code	Y	int	状态码	0
	message	Y	string	返回信息	上传成功
	data	Y	array	成功数据	{ id: 12506 }
请求示例					
	{				
	}				

5 广告管理

- 用于管理设备待机时显示广告信息
- 仅在设备版本 2.3.10 以后版本支持

5.1 设备广告初始化

- 平台配置修改时, 自动获取广告信息

接口用途	获取设备广告信息				
接口地址	http://api.example.com/attendance-advert?device_sn={device_sn}&token={token}				
请求方式	POST				
请求格式	application/json				
头部	参数	必填	类型	说明	值
	X-Requested-With	Y	string	请求类型	XMLHttpRequest
	X-Domain-With	Y	string	请求域	Api
	Authorization	Y	string	验证信息	Bearer + [登陆返回的 token]
请求	参数	必填	类型	说明	示例值
返回	参数	必填	类型	说明	示例值
	success	Y	boolean	状态	true
	code	Y	int	状态码	0
	message	Y	string	返回信息	获取成功
	data	Y	array	成功数据	{}
	➤ <i>title</i>		<i>string</i>	广告标题	促销季
	➤ <i>media</i>		<i>string</i>	媒体文件	http://api.example.com/ads.jpg (大小不超过 10M)
	➤ <i>mediaType</i>		<i>string</i>	媒体类型	图片 image, 视频 video, 网页 webpage
	➤ <i>description</i>		<i>string</i>	广告说明	
	➤ <i>duration</i>		<i>int</i>	播放/持续时长/秒	30

	➤ <i>startDate</i>		<i>string</i>	有效期, 开始日期	2019-01-01
	➤ <i>endDate</i>		<i>string</i>	有效期, 结束日期	2019-09-01
返回					
	<pre>{ "success" : true, "code" : 0, "message" : "", "data" : [] }</pre>				

5.2 下发、调整广告信息至 MQ 设备 (RabbitMQ)

- 平台通过 MQ 下发人员信息到设备
- 终端获取消息后, 根据广告信息调整广告图片
- 照片必须小于 500KB, 最多广告数 10
- 广告请求地址通过 MQ 下发的 path 地址决定, params 会原样 post 至该地址

5.1.1 下发广告更新信息至 MQ

- 平台通过 MQ 下发信息到设备, 下发时以 device_sn 命名队列名称, 设备通过该队列名称获取下发消息。
- 设备根据 method 进行命令识别

信息说明	下发重新获取广告信息指令				
信息用途	更新广告信息				
下发数据	参数	必填	类型	说明	示例值
	method	Y	string	下发命令	sync-advert
	path	Y	string	请求广告信息地址	http://api.example.com/attendance-advert
	notify	Y	string	执行结果通知地址	http://api.example.com/attendance-advert-notify
	params	Y	object	请求 post 数据	{ limit: 100, size: 100 }
下发示例	<pre>{ method: "sync -advert" , path : "http://api.example.com/attendance-advert" , notify: "http://api.example.com/attendance-advert-notify" , params: { limit: 100, size: 100 } }</pre>				
反馈头部	参数	必填	类型	说明	值
	X-Requested-With	Y	string	请求类型	XMLHttpRequest
	X-Domain-With	Y	string	请求域	Api
	Authorization	Y	string	验证信息	Bearer + [登陆返回的 token]
下发反馈	notify 字段如果有值, 则反馈下发状态给 notify 地址				
	<pre>{</pre>				

	<pre> method: "sync -person" , params: {}, success: true, message: "更新成功!" } </pre>
--	---

5.1.2 获取广告信息、更新广告

- 广告更新间隔根据登录返回 advertisePeriod 信息而定，默认只在设备登录是进行请求

接口用途	获取设备广告信息				
接口地址	http://api.example.com/attendance-advert?device_sn={device_sn}&token={token}				
请求方式	POST				
请求格式	application/json				
请求头部	参数	必填	类型	说明	值
	X-Requested-With	Y	string	请求类型	XMLHttpRequest
	X-Domain-With	Y	string	请求域	Api
	Authorization	Y	string	验证信息	Bearer +[登陆返回的 token]
请求	参数	必填	类型	说明	示例值
返回	参数	必填	类型	说明	示例值
	success	Y	boolean	状态	true
	code	Y	int	状态码	0
	message	Y	string	返回信息	获取成功
	data	Y	array	成功数据	{}
	➤ title		string	广告标题	促销季
	➤ media		string	媒体文件	http://api.example.com/ads.jpg (大小不超过 10M)
	➤ mediaType		string	媒体类型	图片 image, 视频 video, 网页 webpage
	➤ description		string	广告说明	
	➤ duration		int	播放/持续时长/秒	30
	➤ startDate		string	有效期, 开始日期	2019-01-01
	➤ endDate		string	有效期, 结束日期	2019-09-01
返回					
	<pre> { "success" : true, "code" : 0, "message" : "", "data" : [] } </pre>				